REMARKS

Status of the Claims

Claims 1, 2, 4, 5, 7, 9-15, 17-22, and 24-26 are pending.

Claim 10 stands objected to due to alleged informality.

Claims 10, 11-19, and 26 stand rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter.

Claims 1-2, 4-5, 7, 9-15, 17-22, and 24-26 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable.

Applicants propose amending claims 1, 10, 11, 20, and 26. Support for the amendments may be found, for example, at ¶¶ 0027, 0048, and 0053. Applicants propose canceling claims 19 and 24.

Summary of Telephonic Interview

The undersigned wishes to thank Examiner Hanh for granting the telephonic interview held June 18, 2008.

During the interview, the undersigned proposed amendments and arguments consistent with those presented herein. Examiner Hanh agreed to give further consideration to the proposed arguments upon submission of a written response.

Rejections Under 35 U.S.C. § 101

Claims 10, 11-19, and 26 stand rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter.

Applicants propose amending claims 10 and 26 to recite a "computer readable **storage** medium." (emphasis added). A computer readable storage medium comprising computer-readable instructions ***is*** patentable subject matter. *In re Beauregard*, 53 F.3d 1583 (Fed. Cir. 1995).

Applicants propose amending claim 11 to recite "a computing processor" and "computing memory communicatively coupled to said computing processor." A system comprising hardware such as recited ***is*** patentable subject matter. 35 U.S.C. § 101.

Reconsideration and withdrawal of the rejections under 35 U.S.C. § 101 is respectfully requested.

Claim Objection

The Office objects to claim 10 due to an alleged informality in the language of the preamble. Applicants propose amending claim 10 remove any alleged informality.

Reconsideration and withdrawal of the objection is respectfully requested.

Rejections Under 35 U.S.C. § 103(e)

Claims 1-2, 4-5, 7, 9-15, 17-22, and 24-26 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. patent publication 2004/0243576 A1 (hereinafter referred to as "Shrivastava") in view of U.S. patent number 6,910,070 (hereinafter referred to as "Mishra"). Applicants traverse the rejection and respectfully request reconsideration.

Applicants have disclosed systems and methods for notifying computing applications that changes in a database have taken place that affect the data that the computing applications retrieve from the database. By way of background, the application explains as follows:

> **[0050]** For every query that is submitted to the database server with a subscription request, the query processor maintains a mechanism that enables it to detect whether a change in the underlying data will affect the result of this query. Such mechanism may comprise a notification manager (as described above). For example, given an email client application that subscribes with a multitude of independent queries like the following ones:
>
> ```
> SELECT name, subject
> FROM users, mail
> WHERE users.id = mail.recipient AND user.name = 'Joe'
> ```

> and
>
> ```
> SELECT name, subject
> FROM users, mail
> WHERE users.id = mail.recipient AND user.name = ‘Jack’
> ```
>
> **[0051]** The notification manager (NM) removes the parameters from the query and stores them in a parameter table of the form as follows:
>
> ```
>    CREATE TABLE parameter_table (param_1
> NVARCHAR(20))
> ```
>
> **[0052]** In the example provided there is one parameter table per query template. The number of columns in the table as well as the type depend on number and type of the parameters in the original query. In this example, the name of a user was given the type NVARCHAR(20). Per subscription, one row with the actual parameter(s) is inserted in the parameter table.
>
> . . .
>
> **[0056]** Using query templates, the change detection query can be formulated as,
>
> ```
> SELECT name, subject
>    FROM
> (SELECT name, subject
> FROM users, mail_delta
> WHERE users.id = mail.recipient) as delta
> JOIN
> parameter_table
> ON delta.name = parameter_table.param_1
> ```
>
> The plan is independent of the number of subscriptions – their individual parameters are stored in parameter_table and are addressed by the join predicate. (Application, ¶¶ 0050 – 0056).

Consistent with this description, amended claim 1 recites a method for providing notifications of changes in a database system comprising:

1. (Currently Amended) A method for providing notifications of changes in a database system, comprising:

receiving a plurality of SQL query statements for querying a relational database system, each SQL query statement corresponding to a computing application that has subscribed to receive notification of changes in the database system affecting data retrieved from the database system by the computing application;

creating a subscription template from the plurality of SQL query statements, the template not comprising any constants;

generating a parameter table from the plurality of SQL query statements, the parameter table comprising for each SQL query statement a constant representing a query value, a subscription identification value uniquely identifying a subscription associated with the particular SQL query statement, and subscriber routing information;

in response to a change in the data in the database, performing a join between said parameter table and said subscription template to generate a query;

executing the query on the database system to identify SQL query statements in the plurality of SQL query statements affected by the change in the data in the database; and

communicating notification to a computing application corresponding to an identified SQL query statement, said notification indicating a change in the data in the database has occurred.

In order for a reference to anticipate claim 1, it must teach the *entirety* of the recited method. The undersigned respectfully submits that neither Shrivastava nor Mishra teach at least the emphasized claim language and cannot possibly teach or even suggest the recited method.

Paragraphs 76 through 86 of Shrivastava, which are referenced by the Office in support of its rejection, disclose a system and method for automatically generating a query statement to search for particular objects or entries in a directory information tree ("DIT") that is stored within relational tables. (*See* Shrivastava, ¶ 76.) Shrivastava teaches using templates to convert an arbitrary lightweight directory access protocol ("LDAP") search filter into a single SQL statement where a base template provides the basic framework for generating an SQL statement. (*Id.* at ¶ 78.) Additional templates are used to fill in specific portions of the base template. The LDAP is converted into a single SQL statement based upon the base template. (*Id.* at ¶ 79.)

Thus, Shrivastava is directed to creating a SQL statement corresponding to an LDAP search filter. In contrast to claim 1, Shrivastava does not disclose a method “for providing notifications of changes in a database system.” Indeed, providing notifications of changes in a database system is not even a consideration of the system disclosed in Shrivastava. Accordingly, Shrivastava does not disclose or suggest the body of the recited claim language.

For example, Shrivastava does not teach or suggest “**receiving a plurality of SQL query statements for querying a relational database system, each query statement corresponding to a computing application that has subscribed to receive notification of changes in the database system affecting data retrieved from the database system by the computing application**.” In fact, Shrivastava does not disclose receiving “**receiving a plurality of SQL query statements for querying a relational database system**” at all. Rather, Shrivastava teaches receiving a **single LDAP search filter**. A single LDAP search filter as disclosed by Shrivastava is not “a plurality of SQL query statements for querying a relational database system” as recited in the claim.

Furthermore, Shrivastava does not teach or suggest “**creating a subscription template from the plurality of SQL query statements, the template not comprising any constants**.” In Shrivastava, the LDAP search filters are received and then **have a template applied to the LDAP filters**. In contrast, the claim language recites “creating a subscription template **from the plurality of SQL query statements**.” Thus, not only does Shrivastava not disclose “receiving a plurality of SQL query statements for querying a relational database system,” but the LDAP search filters that are received in Shrivastava are **not** used to “create a subscription template.”

Furthermore, Shrivastava does not teach “generating a parameter table **from the plurality of SQL query statements, the parameter table comprising for each SQL query statement a constant representing a query value and a subscription identification value uniquely identifying a subscription associated with the particular query statement**.” The Office analogizes that the “catalog tables” mentioned by Shrivastava correspond to the recited parameter table. (Office Action dated 3/31/08, p. 5). But the catalog tables disclosed by Shrivastava do not comprise “for each query statement a constant representing a query value, a subscription identification value uniquely identifying a subscription associated with the particular query statement, and subscriber routing information."

Still further, Shrivastava does not teach "**in response to a change in the data in the database**, performing a join between said parameter table and said subscription template to generate a query." The Office alleges that Shrivastava teaches performing a join between a parameter table and a parameterized subscription template. (Office Action dated 3/31/08, p. 5). For all of the reasons set out in the Reply filed August 13, 2007, Applicant respectfully disagrees with the Office's analysis. Moreover, claim 1 has been amended to recite that the join is performed **"in response to a change in the data in the database."** Shrivastava simply does not disclose performing the recited join "in response to a change in the data in the database." Rather, the join disclosed by Shrivastava that the Office cites to (Shrivastava, ¶¶ 0076-0086) as allegedly teaching this element is made as part of a process for generating a query corresponding to a LDAP search filter and not in response to changes in the data of a database.

Finally, Shrivastava does not disclose or suggest "**executing the query on the database system to identify SQL query statements in the plurality of SQL query statements affected by the change in the data in the database**," and "**communicating notification to a computing application corresponding to an identified SQL query statements, said notification indicating the change in the data in the database has occurred**." As noted above, providing notifications of changes in a database system does not appear to be a consideration of the system disclosed in Shrivastava. Accordingly, Shrivastava does not disclose or suggest "executing the query on the database system to identify query statements . . . affected by the change in the data" and "communicating notification to a computing applications . . . indicating the change in a change in the data in the database has occurred."

Mishra fails to address the deficiencies of Shrivastava. Mishra is directed to a system for asynchronously notifying an application of a database event. (Mishra, Abstract). In the system disclosed by Mishra, a subscription or request to receive notification of a database event is received. (Mishra, Abstract). The request includes a designation of the event name to which the subscription request applies. (Mishra, Abstract). When the occurrence of the event is detected with in the database, a notification of the detected event is published. (Mishra, Abstract).

Thus, Mishra discloses a system wherein a notification request that is received specifies the ***event name*** of which the requesting client wishes to be notified. Thus, Mishra also entirely fails to disclose or suggest "receiving **a plurality of SQL query statements for querying a relational database system**." To the contrary, Mishra teaches receiving a **name** identifying the database event of which the requesting client wishes to be notified. Receiving the name of database event is not the same or even similar to "receiving a plurality of query statements for querying a database system" as recited in the claims.

Furthermore, Mishra does not teach or suggest "**creating a subscription template <u>from the plurality of SQL query statements</u>, <u>the template not comprising any constants</u>**." Indeed, because Mishra does not teach receiving SQL query statements, it cannot possibly teach or suggest "creating a subscription template from the plurality of SQL statements." In fact, the **event names** that are received in the system disclosed by Mishra are not used to create a template.

Still further, Mishra does not teach "generating a parameter table **from the plurality of SQL query statements, the parameter table comprising for each SQL query statement a constant representing a query value and a subscription identification value uniquely identifying a subscription associated with the particular query statement**." Here again, because Mishra does not teach receiving SQL query statements, it cannot possibly teach or suggest "generating a parameter table from the plurality of SQL query statements." Moreover, Mishra does not teach "generating a parameter table" from the event names that are received in the system disclosed by Mishra.

Finally, Mishra does not disclose or suggest "**executing the query on the database system to identify SQL query statements in the plurality of query statements affected by the change in the data in the database.**" Because Mishra does not teach receiving SQL query statements, it cannot possibly teach or suggest "executing the query on the database system to identify SQL query statements in the plurality of query statements affected by the change in the data in the database."

Therefore, because neither Shrivastava and Mishra teach the recited language, the references cannot be combined to form the recited combination of claim 1 and its dependent claims. Similar reasoning applies to independent claims 10, 11, 20, and 26, and all claims depending therefrom.

We note that the claims other than claim 1 recite language that further distinguishes from the cited art. For example, claim 26 recites "processing the SQL queries to generate query templates, **the query templates** not comprising any constants and **being formed as a database table**." The undersigned does not see that cited references disclose or suggest the emphasized language. For this additional reason, claim 26 defines over the cited references.

Withdrawal of the rejections under 35 U.S.C. § 103(a) is respectfully solicited.

CONCLUSION

The undersigned respectfully submits that pending claims are allowable and the application in condition for allowance. A Notice of Allowance is respectfully solicited.

Examiner Thai is invited to call the undersigned in the event a telephone interview will advance prosecution of this application.

Date: June 30, 2008

/John E. McGlynn/
John E. McGlynn
Registration No. 42,863

Woodcock Washburn LLP
Cira Centre
2929 Arch Street, 12th Floor
Philadelphia, PA 19104-2891
Telephone: (215) 568-3100
Facsimile: (215) 568-3439